selecting, in response to the partition figure of merit, a cluster partition to operate as the network cluster; and

halting operation of the remaining cluster partitions.

REMARKS

Claims 1-20 are pending in the application. All claims stand rejected. Claims 1-2, 5-6, 12-14, and 17-18 have been rejected under 35 U.S.C. § 102(e). Claims 3-4, 7-8, 15-16 and 19-20 have been rejected under 35 U.S.C. § 103(a). In response, certain claims have been amended to more distinctly claim the invention. Entry of these amendments will advance the application to issue and narrow the issues for appeal. For the reasons described below, it is believed that all claims are in condition for allowance.

Regarding Claim Amendments

Claims 1, 3-5, 10-13, 15-17, 19 and 20 are amended to further clarify that the Applicants' claimed "application program executing on the network cluster" refers to a "user application program executing on the network cluster." When the ordinary meaning of "application program" is interpreted in the context of the claims in light of the specification, it is believed that one of ordinary skill would interpret "application program" to mean "*user* application program." That is, when the claimed application program interpreted in the context of the invention, it refers to application programs that are directed to application level software architecture, such as a database application, and thus, the claimed application program is not directed to system level software architecture, such as any cluster system software or operating system software. Therefore, the claims are amended to recite "*user* application programs" to specify that this limitation is not directed to system level software architecture. Support for these amendments can be found at least in Claims 1, 3, 7, 10 and 12 as originally filed, and in the Specification at least at page 2, lines 15-19, page 3, lines 18 to 20, page 17 and lines 8-15, and Figure 2 (distributed application 36). Thus, no new matter is being introduced.

In addition, Claims 1, 7, 12, 13, 19 and 20 are amended to further clarify the context of the querying requirement of the claimed invention. In particular, the claims are amended to recite that the *cluster management software determines if a user application program is executing on the partitioned network cluster.* Although this aspect of the invention was implicit

in the claims, the Applicants propose this amendment to ensure proper disclosure and protection of the invention. The claims, as originally filed, recite *querying, by a management program, an application program executing on the partitioned network cluster for the figure of merit*. As such, in order to query the executing user application program, the claimed management program determines if, in fact, there is an user application program executing to query. Therefore, it is believed that this aspect of the claimed invention was implicit. Support for these amendments can be found at least in the claims as originally filed and in the Specification at least at page 3, lines 9 to 21, page 16, lines 25-28, and page 17. Thus, no new matter is being introduced.

Entry of the amendments should not necessitate a new search or place additional burdens on the office. In fact, the distinction between an application program and cluster management software was explicitly described in the Applicant's previous Amendment A (filed October 17, 2002). As such, the claims have been amended to recite how they should have been interpreted. Acceptance and entry of the amendments are respectfully requested.

Regarding Rejections

For the convenience of the Examiner, the Applicants will first address the rejections of independent Claims 1, 7, 12 and 13 and then the rejections of independent Claims 19 and 20.

A. Regarding Rejection of Independent Claims 1, 7, 12 and 13

As recited in Claims 1, 7, 12, 13, 19 and 20, the present invention relates to a figure of merit which indicates a value for a member node of a partitioned network cluster to continue operating. A management program determines if a user application program is executing on the partitioned network cluster, and queries (requests) the user application program for the figure of merit. The user application program determines the figure of merit. The figure of merit is then returned from the user application program to the management program.

Part of the novelty of the present invention is involving a user application program in the determination of a node's value. By having the management program query a user application program, the user application program is afforded an opportunity to provide a vote as to the value of the node upon which it is executing. Thus, the claimed *querying, by a management program, a user application program for the figure of merit* is not taught or suggested by the prior art, as

explained in more detail below. Accordingly, it is respectfully submitted that these claims are allowable.

In the present Office Action, independent Claims 1, 12 and 13 were rejected under 35 U.S.C. §102(e) based on U.S. Patent No. 6,192,401 to Modiri. In a separate rejection, independent claims 1, 7, 12 and 13 were rejected under 35 U.S.C. § 103(a) as unpatentable over Modiri in view of U.S. Patent No. 5,999,712 to Moiin and/or U.S. Patent No. 5,325,526 to Cameron. In these rejections, the Office stated that the Applicants' claimed "application program" corresponded with the software discussed in the Modiri and Moiin references. For example, with respect to Modiri, on pages 2 to 3 of the Office Action, the Examiner states:

> As to querying an application program, Modiri teaches that the method of determining a figure of merit may be implemented in software (column 2, lines 60-62). Modiri teaches that the cluster management software (software layer 250) and an application program (software modules in layer 220) determine a node's value (the software modules in layers 220 and 250 are responsible for determining the membership in the cluster; column 6, lines 30-35) and the cluster framework 220 includes modules such as the cluster membership and quorum and reconfiguration 224 that provides reconfiguration decision making (column 4, lines 50 - 67).

Further, with respect to Moiin, the Examiner states that "Moiin teaches querying (send RECONF_msg to each node) and application (function membership_prop()))."[1]

Although, the Examiner correctly notes that Modiri and Moiin may be implemented in software, it is respectfully submitted that the software implementations discussed in these references correspond to cluster system level software and not to the Applicants' claimed user *application* programs, and, therefore, these references do not disclose the limitations and advantages of the claimed invention.

In conventional systems that resolve cluster partitions, such as those of Modiri and Moiin, cluster software determined a node's value independent of any vote from user applications programs executing on the cluster partition.[2] In the claimed invention, however, the inventors alone recognize the importance in resolving a cluster partition by modifying the conventional

[1] Final Office Action, pg. 7.

[2] *See* Specification, pg. 3, ll. 12-13.

cluster software to enable it to determine if there are any user application programs executing on a node and to query each of the user application programs for a vote on their node's value to the cluster. In this way criteria, such as a number of users and priority of the application can be evaluated when selecting the surviving cluster partition. "This information is valuable as it provides insight into the actual usage of the network cluster."[3] None of the prior art references recognize or discuss the importance of enabling "a *user application*, operating on the cluster, to provide input to the cluster manager. . .regarding the use of the node's resources by the applications actually executing on the nodes."[4]

The Modiri reference, for instance, teaches to use cluster software (cluster management software 220 and cluster framework software 250) to determine a node's value by assigning values to each node, and like the cluster software prior art, these values are determined *independent* of any determining by user application programs executing on the nodes.[5] There is no need in Modiri to involve user application programs in the determination of a node's value because Modiri's invention relates to a different concept that solves different problems with respect to resolving partitioned clusters. In fact, with respect to determining the node's value, Modiri determines a node's value with cluster software regardless of the presence of user application programs. Thus, Modiri does not discuss the advantages or limitations of the claimed invention.

Similarly, the Moiin reference teaches that each node uses its cluster membership monitor to vote on cluster membership and to handle any communication and device failures. Notably, Moiin does nothing to enable *user application* programs to vote on the value of the nodes they are executing upon. Thus, Moiin does not discuss the advantages or limitations of the claimed invention.

Moreover, similar to the Applicants, the Modiri, Moiin and Cameron patents each teach to distinguish between system level software and application programs executing on the network. For example, the Modiri patent discusses that cluster management software and applications

[3] Specification, pg. 3, ll. 13-15.

[4] *See* Specification, pg. 3, ll. 18-21.

[5] *See* Modiri, col. 2, ll. 25-40, and Figure 2.

have distinctly different meanings and distinctly different roles in a distributed system.[6] Likewise, in Moiin, "cluster membership monitor" (CMM) software and application programs are discussed in distinctly different contexts.[7] Also, Cameron specifically distinguishes between applications and system software. For example, Cameron states that:

> *Application programs* (hereinafter applications) comprise processing logic for performing a specific function for a computer user or for the manipulation of a specific type of data. *Applications* are typically **distinguished** from operating system software of which a task scheduler is a part.[8]

Thus, Modiri, Moiin and Cameron each recognize that there are distinct differences between applications running on the nodes and system level software (such as managers, monitors and schedulers), and therefore, it is respectfully submitted that one of ordinary skill would understand that the Applicants' claimed *user application program* is directed to a different level of software architecture and does not correspond to the cluster or system level software discussed in Modiri, Moiin, or Cameron, and obviously, does not correspond to the management software claimed in the Applicants' invention.

Furthermore, Modiri teaches away from the Applicants' claimed invention because all examples in Modiri of criteria used in selecting or favoring member nodes relate to the node's available physical resource or performance contributions to the cluster.[9] More particularly, in determining a node's value, the only factors that Modiri considers relevant are based on what that node can do to increase the power of the cluster and not what the cluster can do for the node. That is, there is no suggestion in Modiri to consider the node's needs, such as the novel idea that

[6] *See* Modiri, col. 2, ll. 26-27 (describing "cluster management software" which assigns weighting values to nodes), and col. 5, ll. 8-33 (describing *applications* (such as "generic applications," "data service applications," and "databases" - which are monitored by the cluster management software for failures/faults)).

[7] *See* Moiin, col. 2, ll. 28-31 (describing that the "CMM is responsible for membership, quorum and failure") and col. 2, ll. 53-55 (describing that "the CMM does not guarantee. . .that the *applications* are present on any given node").

[8] Cameron, col. 1, ll. 20-25 (Emphasis added).

[9] *See* Modiri, col. 2, ll. 11-13 (selecting nodes that can "freely communicate;" col. 2, ll. 35-40: favoring the "fastest nodes"); col. 2, ll. 44-45 (selecting nodes based on "processing power. . .amount of physical memory;" col. 7, ll. 29 (selecting nodes based on "response time to a request"); and col. 9, ll. 10 (selecting nodes that are "functionally operational (i.e. 'healthy')").

the node needs to continue operation because there are user application programs which are executing upon the node.

In addition, the examples of selection factors discussed in Modiri aim to "increase availability and performance" by favoring nodes that have the most processing power and physical memory because these nodes are less susceptible to failure.[10] As such, Modiri teaches to favor nodes that have the most *available* hardware resources to contribute to the cluster, and this is a different concept than the Applicants' claimed invention, which determines a nodes value based on the user applications programs which are actually *consuming* the node's resources. As such, according to the teachings of the Modiri patent, nodes whose resources are consumed by executing user application programs would not be favored because these nodes would not be the "fastest,"as they would likely have less available resources. Thus, by only mentioning factors that provide preference to the fastest nodes, with the most available resources, the Modiri patent teaches away from the Applicants' claimed invention.

Likewise, Moiin teaches away from the Applicants' claimed invention because the criteria used by the cluster membership monitor (CMM) in Moiin to determine members for an new cluster is based on hardware or operating system performance in that it, namely, it is based on the connectivity information (interconnectivity of all the nodes that are directly or indirectly connected).[11] In addition, according to Moiin, "it is also important to note that the CMM does not guarantee the health of the overall system or that the *applications* (emphasis added) are present on any given node. . .[t]he only guarantee is that the system's hardware is up and running and the operating system is present and functioning."[12] Thus, Moiin teaches that the presence of applications on a node is not essential and therefore, Moiin teaches away from the Applicants' claimed invention which requires that a user application program be present to be queried and to determine the value of its node to the cluster.

Furthermore, neither of the references discuss *returning by the user application program the figure of merit to the cluster system manager*. In Modiri, for example, the process of

[10] *See* Modiri, col. 2, ll. 35-40.

[11] *See* Moiin, Abstract, col. 4, ll. 24-40.

[12] Moiin, col. 2, ll. 53-55.

determining a node's value stays within the control of the cluster software. Therefore, Modiri is silent as to any returning of a node's value from applications to the cluster software because the applications do not partake in determining a node's value in the first place. Moiin also does not discuss that an application program returns a node's value to the cluster software because, for example, according to Moiin there is no guarantee that applications are even present.[13]

As none of the Modiri, Moiin and Cameron patents, taken separately or in combination, teach or suggest the claimed "querying, by the cluster management software, a user application program for the figure of merit" and "returning, from the user application program, the figure of merit to the cluster management software" of Applicants' Claims 1, 7, 12 and 13 it is believed these claims and their dependents are allowable.

B. Regarding Rejection of Independent Claims 19 and 20

With respect to the rejections of Claims 19 and 20 based on Modiri and/or Moiin in view of Cameron, the Applicants respectfully submit that the combination of these references does not make the invention obvious.

Among other things, Claims 19 and 20 require that a management program determine if a user application program is executing, and query that user application program for the value of the node upon which it is executing (figure of merit). The Application program assesses merit criteria for determining its execution priority, and determining its nodes value by determining a number of users executing it from its node.

The Applicants respectfully submit that Cameron discloses a typically operating system task scheduler that has been modified to concurrently manage the resources of multiple partitions. Any teachings relating to "priority" in Cameron relate to establishing priority resources allocations for partitions. It is respectfully submitted that there is no motivation in Cameron to combine this "priority" concept to the Modiri and/or Moiin, and furthermore this "priority" concept is not the same as the Applicants' claimed assessing merit criteria for a node to continue operating by determining an execution priority of the user application program, as set forth in Claims 19 and 20.

[13] *See* Moiin, col. 2, ll. 53-55.

The Examiner states on page 6 of the Office Action that, "Modiri as modified by Moiin does not disclose using execution priority of an application to determine figure of merit. However, Cameron teaches (column 11, lines 1-6) that figure of merit (partition priority) can be associated with application priority." It is respectfully submitted that Cameron appears to be nonanlagous art because it relates to a scheduler and allocator that perform resource allocation and partitioning, which includes creating "partition priorities." Nevertheless, if this scheduler and "partition priorities" of Cameron is analogous art, then Applicants respectfully submit that the teaching of "partition priorities" in Cameron are not compatible with the Modiri and Moiin inventions and therefore cannot render the claimed invention obvious.

The Modiri and Moiin patents, for instance, teach to avoid partitioning whereas Cameron's invention is directed to a scheduler and allocator which are "creating new partitions" that have priorities associated with applications.[14] That is, it is an "object" of the Cameron's invention to "provide a multicomputer task scheduler. . . in which a node may be assigned to one or more *partitions* at a time. . .wherein *scheduling priorities of applications* and sub-partitions may be *acquired by the partitions* which contain them. . ."[15] In comparison, for example, the Moiin patent seeks to resolve cluster partitions by providing a "generalized algorithm that deals with the issue of a partitioned network."[16] Further, according to Modiri, "[t]he *split-brain condition* leads to data and file corruption," column. 2, lines 5-16, and "[b]asing cluster membership decisions upon weighting factors . . .may advantageously increase availability and performance by favoring the most valued (fastest, etc.) nodes. . .to *prevent split-brain* configurations," column 2, lines 35-40.

Thus, Cameron teaches away from Modiri and Moiin by creating and enabling partitions whereas Modiri and Moiin seek to eliminate partitions (split-brain conditions and the like). In view of Cameron's teachings to create partitions and Modiri and Moiin's teachings to eliminate them by voting on or selecting surviving partitions, it is respectfully submitted that the

[14] *See* Cameron, col. 3, ll. 17-20, and Abstract.

[15] *See* Cameron, col. 3, ll. 62 - col. 4, ll. 1.

[16] *See* Moiin, col. 1, ll. 57-60.

combination of the teachings in Cameron with Modiri and Moiin would be inoperative. That is, the partition creation teachings of Cameron frustrate the purpose of Modiri and Moiin.

Furthermore, on page 6 and 8 of the Office Action, the Examiner states that "[i]t would have been obvious to apply associating application priority with figure of merit as taught by Cameron to the invention of Modiri as modified by Moiin because using application priority to determine the importance of a partition to continue operating would insure that applications with high priorities would continue processing." The Applicants respectfully submit, that there is nothing in any of the references to suggest "*using application priority to determine the importance of a partition to continue operating*" because none of the references suggest a connection between an application's execution priority and the importance a cluster partition to continue operating.

In addition, there is nothing in any of the references to suggest that this "*would ensure that applications with high priorities would continue processing.*" Cameron, for instance, does not discuss anything about the fact that there is a possibility that any *application would not continue processing*. In addition, the discussions of criteria used in selecting nodes in Modiri and Moiin are all related to the resource contribution, hardware performance or health of the nodes, and therefore, they actually teach away from the idea that the executing priority of a user application or mere presence of user application should be a factor at all in selecting nodes to continue operation.

Thus, it respectfully submitted that there is no motivation to combine these references, and that the only teaching cited by the Examiner regarding the relative usefulness of the Applicants' claimed *assessing merit criteria for the value of a node by determining the execution priority of the application program* is taken from the applicants own patent disclosure.[17] It is submitted that the citation of the Applicants own application, for teachings not found in any of the prior art references, constitutes impermissible hindsight-based obviousness analysis. The teaching or suggestion to make the claimed invention must come from the prior art, not from the Applicants' own disclosure.[18]

[17] *See* Final Office Action, pgs. 6 and 8.

[18] *See In re Vaeck*, 947 F.2d 488 (Fed. Cir. 1991).

With respect to the claim limitation of assessing merit criteria by determining the number of users executing from the node, as set forth in Claims 19 and 20, it is respectfully submitted that the combination of these references does not make this aspect of the invention obvious. Specifically, on page 7 of the Final Office Action, the Examiner states that this would have been obvious "because the number of users executing on the node would determine the processing load of each node," and the Applicant respectfully submits that none of the references render *determining a number of users executing from the node* obvious. Although Modiri considers the processing power of a node a relevant factor in the determination of that node's value, there is no suggestion to determine a *number* of users executing from the node to establish the processing power of a node.

Moreover, on page 6 of the Office Action, the Examiner stated that "Cameron teaches (column 9, lines 22-37) determining (manage) the number of users (a list of consumers) executing from the node. . ." The Applicants respectfully submit that the consumers discussed in Cameron's are not *users*. That is, in Cameron, "Consumers are either application programs or other partitions (i.e. sub-partitions)."[19] Thus, Cameron does not suggest determining the number of users on a node.

As none of the prior art teaches or suggest Claim 19 and 20, it is believed that these claims are allowable.

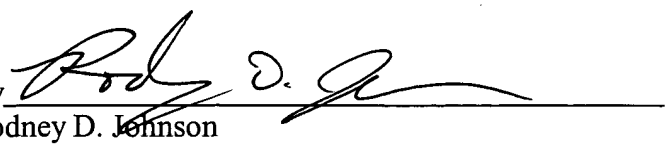
[19] Cameron, col. 9, ll. 23-26.

CONCLUSION

In view of the above amendments and remarks, it is believed that all claims are in condition for allowance, and it is respectfully requested that the application be passed to issue. If the Examiner feels that a telephone conference would expedite prosecution of this case, the Examiner is invited to call the undersigned attorney at (978) 341-0036.

Respectfully submitted,

HAMILTON, BROOK, SMITH & REYNOLDS, P.C.

By 

Rodney D. Johnson
Registration No. 36,558
Telephone: (978) 341-0036
Facsimile: (978) 341-0136

Concord, MA 01742-9133

Dated: April 3, 2003



MARKED UP VERSION OF AMENDMENTS

Claim Amendments Under 37 C.F.R. § 1.121(c)(1)(ii)

1. (Amended) A method for providing a figure of merit indicating a value for a member node of a partitioned network cluster to continue operating, the method comprising:

determining, by a management program, if a user application program is executing on the partitioned network cluster;

querying, by [a] the management program, the [an] user application program [executing on the partitioned network cluster] for the figure of merit;

determining, by the user application program, the figure of merit; and

returning the figure of merit from the user application program to the management program.

3. (Amended) The method of Claim 2 wherein assessing merit criteria includes:

determining the number of users executing the user application program from the member node.

4. (Amended) The method of Claim 2 wherein assessing merit criteria includes:

determining an execution priority of the user application program.

5. (Amended) The method of Claim 1 wherein querying the user application program includes:

providing a proposed figure of merit.

7. (Amended) A method for resolving a partitioned computer network cluster including multiple cluster partitions, the method comprising:

evaluating a partition figure of merit for each cluster partition including:

determining that a user application program is executing on the cluster partition at a member node from a member node;

requesting, from [an] the user application program [executing on the cluster partition,] a node figure of merit, the figure of merit indicating a value to the user

application program for [a] the member node [on which the application is executing,] to continue operation;

providing, from the user application program, the requested node figure of merit; and

evaluating the provided node figure of merit to determine to partition figure of merit;

selecting, in response to the partition figure of merit, a cluster partition to operate as the network cluster; and

halting operation of the remaining cluster partitions.

8. (Amended) The method of Claim [1] 7 wherein requesting a node figure of merit further includes:

providing, from a cluster manager executing on the member node, a proposed node figure of merit.

10. (Amended) The method of Claim 9 wherein assessing merit criteria includes:

determining a number of users executing the user application program from the member node.

11. (Amended) The method of Claim 9 wherein assessing merit criteria includes:

determining an execution priority of the user application program.

12. (Amended) A computer program product for providing a figure of merit indicating a value for a member node of a partitioned network cluster to continue operating, the computer program product comprising a computer usable medium having computer readable code thereon, including program code which:

determines if a user application program is executing on the partitioned network cluster; and

queries [a] the user application program executing on the partitioned network cluster for the figure of merit.

13. (Amended) A system for providing a figure of merit indicating a value for a member node of a partitioned network cluster to continue operating, the system comprising:

a means for determining, by a management program, that an application program is executing on the partitioned network cluster;

a means for querying, by the management program, the [an] user application program [executing on the partitioned network cluster] for the figure of merit;

a means for determining, by the user application program, the figure of merit; and

a means for returning the figure of merit from the user application program to the management program.

15. (Amended) The system of Claim 14 wherein a means for assessing merit criteria includes:

a means for determining a number of users executing the user application program from the member node.

16. (Amended) The system of Claim 14 wherein a means for assessing merit criteria includes:

a means for determining an execution priority of the user application program.

17. (Amended) The system of Claim 13 wherein a means for querying the user application program includes:

a means for providing a proposed figure of merit.

19. (Amended) A method for providing a figure of merit indicating a value for a member node of a partitioned network cluster to continue operating, the method comprising:

determining, by a management program, if a user application program is executing on the partitioned network cluster from the member node;

providing a proposed figure of merit from a management program to [an] the user application program [executing on the partitioned network cluster];

querying, by the management program, the user application program [executing on the partitioned network cluster] for the figure of merit;

assessing, by the user application program, merit criteria for the figure of merit, wherein assessing merit criteria including:

determining a number of users executing the user application program from the member node; and

determining an execution priority of the user application program;

determining an alternate figure of merit derived by assessing the merit criteria;

selecting, by the user application program, between the proposed figure of merit and the alternate figure of merit; and

returning the figure of merit from the user application program to the management program.

20. (Amended) A method for resolving a partitioned computer network cluster including multiple cluster partitions, the method comprising:

evaluating a partition figure of merit for each cluster partition including:

determining, by a cluster manager, that a user application program is executing on the cluster partition from a member node;

providing, from [a] the cluster manager [executing on the member node,] a proposed node figure of merit to [an] the user application program [executing on the cluster partition];

requesting, from the user application program, a node figure of merit, indicating a value to the user application program for [a] the member node, on which the application is executing, to continue operation;

assessing, by the user application program, merit criteria for the member node, wherein assessing merit criteria including:

determining a number of users executing the user application program from the member node; and

determining an execution priority of the user application program;

determining, by the user application program, an alternate node figure of merit derived by assessing the merit criteria for the member node;

selecting between the proposed node figure of merit and the alternate node figure of merit;

providing, from the user application program, the requested node figure of merit; and

evaluating the provided node figure of merit to determine the partition figure of merit;

selecting, in response to the partition figure of merit, a cluster partition to operate as the network cluster; and

halting operation of the remaining cluster partitions.